Can Intermediaries in Programmatic Advertising Obtain Economical Benefit from Invalid Traffic Filtering!? Why and How?

Anonymous Author(s)

ABSTRACT

Invalid traffic is an inherent problem of programmatic advertising and has not been properly addressed so far. Traditionally, it has been considered that invalid traffic only harms the interests of advertisers, which pay for the cost of invalid ad impressions while other industry stakeholders earn revenue through commissions regardless of the quality of the impression. Our first contribution consists of providing solid evidence that shows how the Demand Side Platforms (DSP), one of the most important intermediaries in the programmatic advertising supply chain, are indeed suffering from economic losses due to invalid traffic. To solve this problem, DSPs require a highly scalable solution that is able to identify invalid traffic at the level of individual bid requests, in real-time without adding other than negligible cost. The second contribution is the design and implementation of such a solution to be integrated in the current programmatic ecosystem by the DSPs. The detection algorithm of this system leverages the concept of Shannon entropy for identifying domains with anomalous traffic patterns associated with invalid traffic. The third contribution of this paper is proposing a paradigm shift towards a more transparent approach for solving the invalid traffic identification problem. We advocate for the need of defining open-source invalid traffic detection techniques, and thus have made the code of our system publicly available under open-source license and are in active discussions with various stakeholders in the Adtech industry to get our solution widely adopted by DSPs.

1. INTRODUCTION

In the $700 billion per year global media investment market [63], nearly 25 % is invested into digital media formats such as banner [27, 57], video and in-app ads. Once a visitor leaves a webpage, the opportunity to place an ad in front of the user perishes. As a result, advertising sell-side is incentivized to adopt strategies with the goal of monetizing every impression opportunity, every time. In contrast, the buyer (advertiser) has the preference for buying the impressions with the highest possible marketing effect. In the current programmatic media market, described in detail in Section 2, the buyer and seller are typically separated by various middle-men. Each is compensated by a commission on the basis of the transactions they are part of. It has been reported that 55 % or more of the buyer's investment goes into paying commissions [9] of various middle-men. This conflict of interest has significantly contributed to the growth of the invalid traffic problem [2], which has not been properly addressed so far. Traditionally, it has been considered that invalid traffic only harms the interests of advertisers, which pay for the cost of the invalid ad impressions. Intermediaries in the supply chain get a commission for each served invalid impression and they do not have direct monetary incentives to effectively fight invalid traffic.

Specialized companies referred to as verification vendors (e.g. IAS [28], DoubleVerify [19], Whiteops [61]) have emerged to address this specific issue. Verification vendors (and other players) implement opaque proprietary solutions for the identification of invalid traffic. These vendors argue that opacity is needed to avoid providing valuable information to potential fraudsters, but previous research has shown that even simple attack vectors can defeat these opaque defenses [16, 38]. In addition, opacity prevents the possibility of independent auditing of these detection techniques. Similar traffic anomaly detection problems in related areas such as network intrusion detection have been addressed based on transparent, open-source solutions such as Snort [47] or Bro [53].

In this paper, we propose a novel solution for invalid traffic detection and filtering in programmatic advertising. Our first contribution consists of providing solid evidence that shows how Demand Side Platforms (DSPs) are suffering from economic losses due to invalid traffic. This result refutes the general assumption that intermediaries profit from invalid traffic and provides DSPs an evidence based model for evaluating the effect of invalid traffic on the economics of their business model.

Our hypothesis is that post-bid (i.e. non real-time) detection of invalid traffic does not solve the problem for the DSPs. Instead, DSPs require a solution that is able to identify invalid traffic in real-time and at the level of individual bid requests. DSPs handle up to tens of billions of bid requests per day, a factor imposing demanding computational performance constrains to the invalid traffic detection problem.

As our second contribution, we solve this problem by the design and implementation of an affordable system that we refer to as *Nameles*, which identifies anomalous traffic patterns of domains using an algorithm based on Shannon entropy. Nameles has been built in accord with the latest version of openRTB specification [34] and is able to handle up to 500 k bid requests per second, adding a total delay of 3ms or less to each bid request. As a result, it can be seamlessly integrated in to the programmatic supply-chain as a solution for the DSPs.

Finally we propose a paradigm shift towards transparency from the current opacity based invalid traffic detection and filtering approach. While the current opaque approach has been shown flawed [16, 38], open-source software has been proven a key success factor in other related areas, e.g. Snort [47] for Network Intrusion Detection. Based on these experiences, a prototype of Nameles has been released as the first open-source solution for invalid traffic classification. Our aim is that this prototype serves as a platform for bringing together contributors from academia, the Adtech industry, and the Information Security industry, in order to create the future of invalid traffic detection and filtering as a community effort. Along these lines, the main global advertiser trade-body World Federation of Advertisers (WFA) has advocated for the need of open-source solutions and has specifically endorsed Nameles.

2. BACKGROUND

The Ad Exchange (also referred to as the SSP), the Demand Side Platform (DSP), the media agency, and verification vendors are the main intermediaries in the programmatic advertising ecosystem. The Ad Exchange aggregates inventory from up to tens of thousands of publishers, and the DSPs each connect to up to a 100 Ad Exchanges. Advertisers and their media agency partners use DSPs to programmatically bid on media available in the Ad Exchanges. The main role of the verification vendors is to help DSPs filter out poor quality traffic, such as invalid traffic as defined by Media Rating's Council in its Invalid Traffic Filtration Guidelines [14]. Each intermediary receives a commission for its participation in an ad transaction. The money flow among these players is depicted in Figure 1.

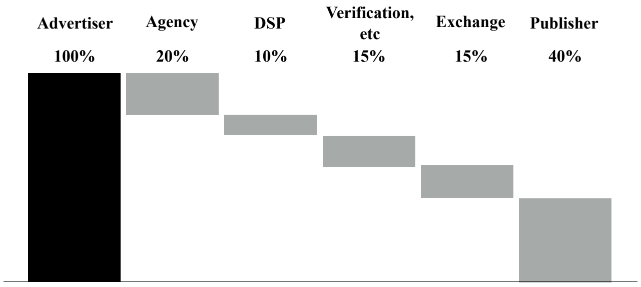

Figure 1: (Left-to-Right) Money flow in the programmatic advertising ecosystem.

In the recent years advertisers have become increasingly vocal about their concerns related with the quality of programmatic media transactions [21, 54, 59], and the lack of transparency in the ecosystem [10, 17]. Out of the key intermediaries, the DSP plays a central role in protecting the interest of the advertiser.

DSPs connect media buyers with Ad Exchanges, which on behalf of their publisher partners send bid auction requests to the DSPs. Each bid represents an opportunity for the DSP to match demand on the buy-side with supply on the sell-side. In effect each bid event corresponds with an opportunity to place an online advertisement on a webpage for the advertiser, and an opportunity to monetize an ad placement for the publisher. Based on the respective commission percentages, the intermediaries are compensated every time a bid is successfully transacted and an ad is displayed as a result. However, the advertiser benefits only when the traffic associated with the transaction is valid.

According to various industry guidelines [58, 59], invalid traffic is defined to correspond with those bid events where displaying an ad would not have any potential for advertising effect and the advertiser would lose its investment without getting anything in return. Various industry bodies and committees of established bodies have been created to focus on the invalid traffic problem: JICWEBS, TAG, Botlab, and MRC's Invalid Traffic Committee [7, 13, 32, 55].

3. DATASET

The dataset used in this paper includes a daily sample of incoming bid-stream data collected between December 01, 2016 and December 25, 2016. The data is from one of the largest DSPs with significant global presence. The data consist of desktop and mobile bid events, for video, banner and in-app inventory. In particular, each daily sample includes between ~1.7 - ~1.9 Billion actual bid requests issued on that date from ~50 Ad Exchanges. These bid requests are associated (in average) to ~150 M IP addresses and ~900 k domains per day. The dataset includes the following information per bid request: a unique identifier, the IP address of the device initiating the bid event and the Web Domain or Mobile Application ID selling the ad space. Note that for simplicity we refer to both Web Domains and Mobile Applications as *Domains* along the paper.

4. ECONOMIC IMPACT OF INVALID TRAFFIC IN DSPS

In this section we refute the argument that advertisers are the only stakeholders in the programmatic ecosystem negatively affected by invalid traffic [2] by providing a detailed economic model that demonstrates how invalid traffic negatively impacts the cash flows of DSPs.

DSP companies are rarely profitable [6] and consequently are dependent on external investment to sustain their business. We investigated seven publicly listed DSPs through their annual income statements and found that only one company had a positive net income [6]. Depending on the DSP, IT costs ranged from 30 % to 50 % of the revenue [6]. These findings show that the current operation of major DSPs creates systematic losses, and that losses have strong correlation with IT costs. We also conclude that the investor sentiment is shifting against DSP companies during the last two quarters of 2016, further complicating the economic position such companies have in the programmatic market [6, 37].

4.1 Increasing DSP Valuation and Profitability Through Traffic Filtering

The DSP win-rate [17], the fraction of won bids out of all auctions, indicates how much the DSP is paying for IT resources that yield no revenue. Therefore the cash flows of a DSP company are closely related with its win-rate. Regardless if an auction the DSP is hosting results in a win or not, the DSP bears the IT cost for facilitating that auction. Having interviewed leading DSPs, we conclude that the DSP win-rate is between 10 % to 20 %. An individual advertiser win-rate has been shown to be in the range 0.1 % to 1 % [65] and an ad exchange fill-rate in the range 10 % to 40 % [46], and further infer that the DSP win-rate will be between the two. Consequently, we posit that there is a significant over-supply of programmatic media impressions, which supports the economic viability of invalid traffic filtering. If a given DSP's win-rate is lower than the fraction of bids filtered out, in theory, there is no loss of economic opportunity for the DSP. In addition to improving profitability and valuation of a DSP, filtering invalid traffic reduces strategic risk associated with undisclosed exposure to ad fraud. In the case of two DSP companies [33, 52], each lost significant fraction of their market capitalization as a direct result of their exposure to invalid traffic becoming evident to investors. Because of the wide concerns [21, 31, 35, 54] with invalid traffic, using a transparent method for filtering may also increase the credibility a given DSP has in the eyes of the market. In order to maximize the benefits for the DSP, we further suggest that the DSP has to implement filtering at pre-bid stage in real-time.

4.2 Economics of the Demand Side Platform Business Model

Net Present Value (NPV) model is the tool of choice for financial forecasting because it considers the time value of money, and provides a concrete metric to financial decision makers, such as investors, for evaluating investment against the predicted return [8]. Finance theory endorses an investment if NPV is positive and higher than NPV of an alternative investment [8]. In addition to the NPV, we evaluated Enterprise Value (EV) [29], a useful variant of the NPV, that takes into account cash flows beyond the forecasted time window. Positive NPV and EV values can be reached when the cash inflows exceed cash outflows [8]. NPV and EV are widely used as decision-making tools for planning purchases, mergers or acquisitions [8].

In our model we compared two scenarios; without invalid traffic filtering (Scenario A), and with filtering using a solution similar to the one proposed in Section 5 (Scenario B), in a timeframe of 8 years. NPV and EV models require 5 key factors for presenting the outputs: 1) annual growth rates, which are based on industry average of seven publicly listed DSPs' annual and quarterly income statements between 2012-2015 [24], 2) a typical rate of return r for investments made into new systems or products is 20 % [8, 30], 3) a variable invalid traffic filtering rate F, 4) revenue penalty P as a dependent factor of F, and 5) a long-term cash flow growth rate G of 2 % [36]. Both scenarios A and B have the same r and G values.

We have selected the parameters of the penalty function to make the penalty increase in an exponential manner, such that the penalty is low until $F = 20-30\,\%$ and it spikes after this point. These percentages correspond to the average reported fraction of invalid traffic from different studies [18, 64] as well as insights from the industry. Even in cases where the ideal filtering rate would be lower, the model provides evidence that there will still be significant economic gain for DSPs as a result of filtering invalid traffic.

Table 1 presents the range of values for NPV and EV of 7 DSPs and the model discussed above. In particular, the table shows minimum and maximum of data points in group $F\,\%$ and $P\,\%$ combined to analyze the effect on NPV and EV. $Max[F, P] = Max[0.23, 0]$ and $Min[F, P] = Min[1, 1]$ for all DSPs. Both EV and NPV values give promising results supporting the argument that DSPs can gain significant economic benefits from pre-bid invalid traffic filtering at an appropriate filtering rate.

As a result of the increasing win-rate 1) the price required to win bids grows higher than buyer algorithm ceiling prices allow bidding for, and 2) there is more demand than available inventory. The above mentioned factors result in a drop of DSP revenue, consequently

	Enterprise Value			**Net Present Value**		
	No filtering	Filtering		No filtering	Filtering	
		Max [F,P] range	**Min [F,P] range**		**Max [F,P] range**	**Min [F,P] range**
DSP-1	10.544	19.002	-3.269	4.421	8.020	-1.979
DSP-2	2.516	5.194	-3.518	536	1.672	-2.213
DSP-3	3.254	3.833	-1.147	1.237	1.481	-706
DSP-4	1.184	2.973	-2.376	133	892	-1.498
DSP-5	1.702	2.648	-819	634	1.035	-506
DSP-6	1-354	2.342	-1.005	445	863	-628
DSP-7	1.595	2.262	-2.118	310	592	-1.337
Industry avg. **ACME**	3.163	5.468	-2.036	1.101	2.079	-1.267

Table 1: Impact of invalid traffic filtering to economics of DSPs.

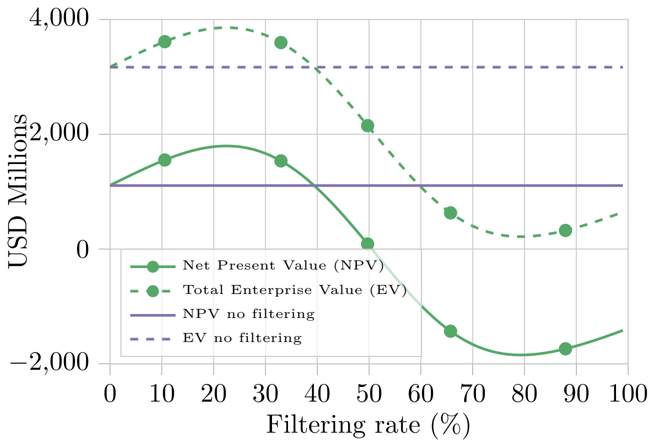

Figure 2: A combined sensitivity analysis of invalid traffic filtering rate F and win rate W and their effect on NPV.

leading to a negative effect on NPV and EV in comparison to no filtering at all. Therefore, it is vital for the DSP to find the right balance between filtering rate and win-rate to achieve optimal benefits on attractiveness (NPV), valuation (EV) and profitability. Figure 2 shows the results of the sensitivity analysis of F (filtering rate) and P (revenue penalty) where the filtering and the penalty rates impacts only on the scenario B. To perform this analysis, we have created ACME Inc. as a representative example of a DSP using average values of the 7 DSP companies.

We observed that there are NPV and EV gains for the DSP when filtering rate increases from zero towards $F = 23\,\%$. Filtering invalid traffic beyond $F > 23\,\%$ first results in diminishing benefit and eventually drives a decline in revenue for the DSP. Benefits start to actualize immediately once the traffic filtering is activated. Results indicate that filtering invalid traffic requires ongoing monitoring on behalf of the DSP for establishing ideal filtering rates.

4.3 Validity of Results

We computed a sensitivity analysis for four key inputs in the model; 1) rate of return r, 2) long-term growth rate G, 3) traffic filtering rate F, and 4) revenue penalty P. Sensitivity analysis for r, ceteris paribus, consisted of group of data points $r(\%) = 10, 15, 20, 25, 30$. Results show that increasing r 5 %, increases NPV 31 % and EV 69 % on average for the scenario B. Sensitivity analysis for G consisted of a group of data points $G(\%) = 1, 1.5, 2, 2.5, 3$. Increasing G by 0.5 % increased EV 4.7 % in scenario B on average.

In the economic model, we have demonstrated the impact of filtering invalid traffic based on the actual financial data from DSPs and realistic inputs for computing both NPV and EV. We suggest that by utilizing invalid traffic filtering methods such as the one described in Section 5, the DSP's total costs can be decreased significantly without compromising gross revenue, resulting in an improved NPV and EV for the DSP.

5. SYSTEM REQUIREMENTS, DESIGN AND IMPLEMENTATION

In this Section, we define a system that addresses the issues highlighted in the previous sections and refer to it as *Nameles*. We will first describe the fundamental operational requirements of the system and then provide details on its design and implementation.

5.1 System Requirements

Our system has the following key functional requirements:
1. Scalability: DSPs typically handle tens of billions of bid requests per day. This maps into peaks of hundreds of thousands bid request per second, and the system must be capable of handling these high rates of bid requests.
2. Delay: The bid response to a given bid request has to be received by the Ad Exchange within 100 ms [22]. Hence, the delay introduced should be limited to few ms in order to minimize the impact in the overall bidding process delay.

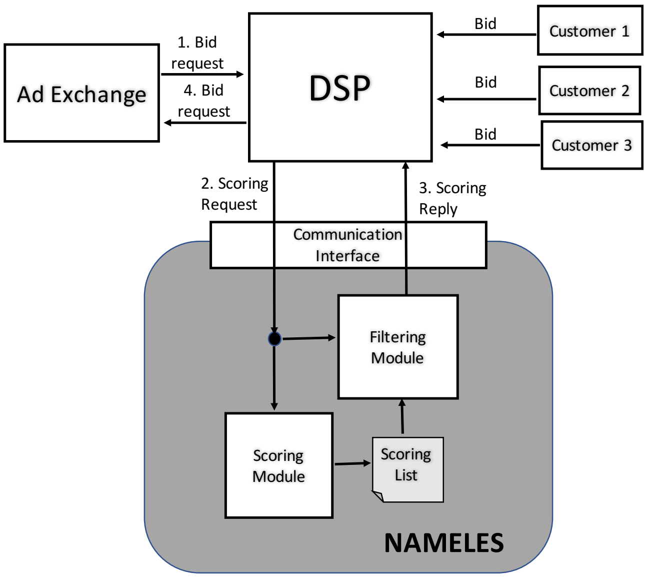

Figure 3: Programmatic Ecosystem Scheme with Nameles Implemented.

3. Accuracy in invalid traffic identification: Providing 100 % guarantee that a bid request is invalid (or not) is not feasible. Instead, it is more reliable providing a Confidence Score associated to a bid request indicating the likelihood that such bid request is invalid. Therefore, our system must incorporate an accurate scoring algorithm.

5.2 System Design

In this subsection we first present a brief overview of the system functionality and how it is integrated within the programmatic advertising ecosystem and more specifically with the DSPs. Then, we describe in detail the functional blocks forming the Nameles system: the Communication Interface, the Scoring Module, and the Filtering Module.

5.2.1 Overview

Figure 3 presents a high level representation of Nameles functional blocks. Moreover, the figure shows how Nameles could be integrated in the programmatic ad delivery chain as an auxiliary service for the DSPs. The only difference with respect to the current operation of a DSP would be that, as part of the pre-bid phase, the DSP makes a request to Nameles to provide a Confidence Score per bid request. To this end, the DSP sends a *scoring request* to Nameles (step 2 in Figure 3). The scoring request includes the following fields: bid request id (mapping Nameles result to the corresponding bid request), IP address of the device associated with the bid event and the domain offering the ad space. This information is included in the bid requests as defined in the openRTB protocol standard [34]. The *scoring request* is delivered to two independent modules of Nameles: the *Scoring* module and the *Filtering* module.

Because the DSP has limited information about a bid request to determine if it is invalid or not, we propose to aggregate all bid requests from a domain and use statistical analysis to determine the level of confidence of a domain. This approach provides statistically robust Confidence Scores for domains since they are computed from a sample of (at least) hundreds of bid requests. Then, Nameles assigns to the bid requests from a domain the Confidence Score of such domain. The Scoring Module is responsible for computing the Confidence Score for domains present in the bid requests received by the DSP. Moreover, it groups the domains in four different Confidence Classes. The traffic profile associated with a given domain may change significantly over time, resulting in a higher (or lower) confidence. To address this issue, the Scoring Module recomputes the Confidence Score of each domain every day. As a result of the described process, the Scoring Module produces every day a *Scoring List* that includes both the Confidence Score and the Confidence Class for each individual domain.

The *Filtering* module is responsible for classifying in real-time each received *scoring request*. To this end, it retrieves the domain id from the *scoring request* and obtains the domain's Confidence Score and Confidence Class from the *Scoring List* introduced above. After that, it creates a *scoring reply* to be sent to the DSP (Step 3 in Figure 3). This reply includes the following information: bid request id (extracted from the corresponding scoring request), the domain Confidence Score, and the domain Confidence Class. If the domain is not present in the Scoring List, the scoring reply includes NULL values for the Confidence Score and the Confidence Class.

Finally, the communication between the DSP and Nameles is handled by the *Communication Interface Module*.

5.2.2 Communication Interface Module

This module is responsible for handling the communication between the DSP and Nameles. Specifically, it manages the delivery of scoring requests from the DSP to Nameles and scoring replies in the opposite direction. We have opted to use a parallel pipeline communication structure as depicted in Figure 4. In particular, the DSP creates two queues: a sending queue used for pushing scoring requests to Nameles and a receiving queue for pulling scoring replies from Nameles in return. Nameles sets up a number of worker processes, which connect to the sockets associated with both queues. These workers pull scoring requests from the sending queue and forward them to the Scoring and Filtering modules. The

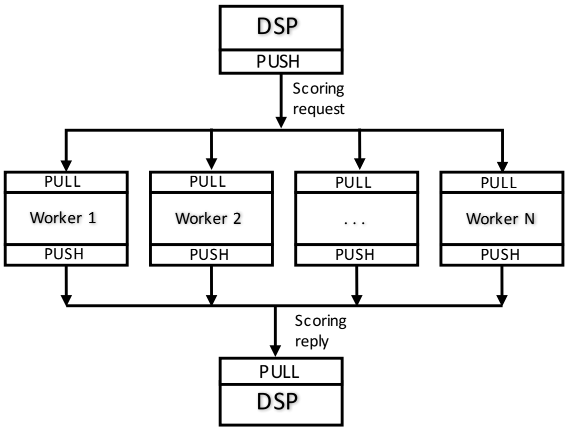

Figure 4: Parallel Pipeline communication structure.

result of the filtering process is pushed by the workers to the receiving queue of the DSP.

The parallel pipeline communication structure offers a number of characteristics that make it a suitable solution in our case. First, it is easy to implement, thus requiring a low deployment effort for the DSPs using Nameles. Second, it offers outstanding scalability performance, being able to handle streams of hundreds of thousands requests per second with processing delays below 3 ms. Third, it can be implemented using existing message handling solutions and middleware [3, 26, 51].

5.2.3 Scoring Module

The goal of the scoring module is to produce a *Scoring List* of domains to be used by the *Filtering* module. This list is updated daily. Since Nameles operates in real-time, the list used at day d is obtained from a prediction algorithm applied on the historical *Confidence Score* values of domains at days $d-1$, $d-2$, $d-3$, ...

To produce the *Scoring List*, the Scoring Module implements 3 different algorithms: one to compute the Confidence Score of each individual domain, a second to compute the Confidence Classes, and a third to derive the *Scoring* list to be used at day d based on historical information. Next we describe each of these algorithms.

- Confidence Score computation: A DSP can use the bid requests associated with a domain to reconstruct its traffic pattern based on IP addresses associated with the bid requests. This is the fundamental signal used by our algorithm. Skewed distributions, where most bid requests come from just few IP addresses, are for obvious reasons suspicious[1] and thus domains presenting such traffic patterns should be assigned low Confidence Scores. Instead, legit traffic patterns correspond to more homogeneous distributions of bid requests across IPs and domains presenting such distributions should receive high Confidence Scores.

We compute the Shannon Entropy [44] of the distribution of bid requests across IP addresses for each domain in the considered dataset. The Shannon Entropy summarizes in a single value the level of determinism of a distribution and ranges between 0 (all bid requests to a domain come from a single IP address) and $log_2(n)$ (the bid requests are homogeneously distributed across the n IP addresses making ad requests to the domain). We use the following expression to compute the Entropy ($H(X)$) for a domain X:

$$H(X) = log_2(C(X)) - \frac{\sum_{i=1}^{n} C(x_i) log_2(C(x_i))}{C(X)} \tag{1}$$

where, $C(x_i)$ represents the number of bid requests received by the domain from IP$_i$, and $C(X)$ represents the total number of bid requests associated with the domain.

Shannon entropy has been successfully used in a wide range of applications [44]. However, in our case, it has an important limitation because it does not consider the volume of bid requests, but just the shape of the distribution of bid requests. This avoids making direct comparison of domains with different volumes of bid requests. For instance, a domain with 5 bid requests uniformly distributed across 5 IPs would have the same Entropy value (2.32) than a domain with 5000 bid requests homogeneously distributed across 5 IPs. While the first domain is just an unpopular domain, the second one is highly suspicious, having a high number of daily visits from a small number of IPs distributed evenly.

To address this limitation, we propose a simple normalization process that takes into account the volume of bid request associated to a domain. In essence, we compute the ratio of the entropy ($H(X)$) and the binary logarithm of the total number of bid requests ($C(X)$) and scale the resulting value to a normalized range between 0 and 100. This normalized entropy score is the *Confidence Score* (CS) assigned to domains by Nameles and its formal expression is:

$$CS(X) = 100 \left(1 - \frac{\sum_{i=1}^{n} C(x_i) log_2(C(x_i))}{C(X) log_2(C(X))}\right) \tag{2}$$

To get an intuition on the effect of this normalization process, we can consider the simplistic example mentioned above. In this case, the domains with 5 bid requests from 5 IP address would have a high CS equal to 100 whereas the domain with 5000 bid requests would have a low CS equal to 19.

- Computation of the Confidence Classes: We first analyzed the probability distribution function of the CS values across domains in our daily datasets. Fig-

[1] For instance, this can be the result of a domain receiving most of its visits from scrapers or from other types of bots associated with invalid traffic.

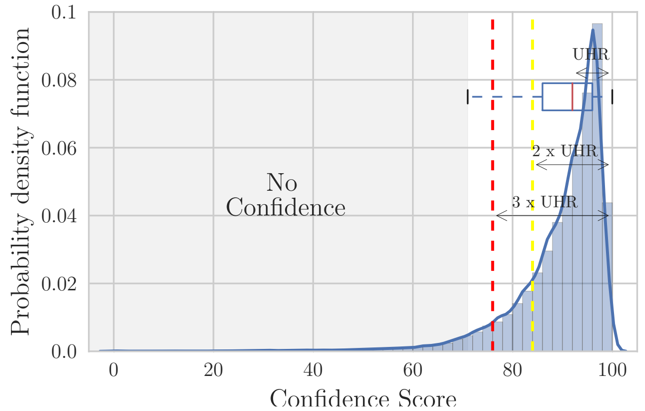

Figure 5: Distribution of Confidence Score (CS) values for domains with more than 500 bid requests at December 1, 2016.

ure 5 shows this distribution for a specific day. Note that other days in our dataset showed similar distributions. We observed a skewed distribution concentrated in the high CS values with a long tail towards low CS values. This indicates that most domains present homogeneous traffic patterns (represented by high CS) whereas as we move towards low values less domains are found presenting increasingly deterministic patterns. In other words, as we move towards lower values of CS we find domains with infrequent (i.e., statistically unlikely) traffic patterns offering lower confidence.

To define the Confidence Classes, we use two different unsupervised statistical methods that divide the distribution in 4 ranges each representing a single Confidence Class:

- *Outlier detection method*: This method identifies outlier CS values based on the definition of traditional outliers [42], i.e., $CS(X) < 25\ percentile - 1.5 \times IQR$. Nameles uses this expression to define the threshold for the *No Confidence* Class including domains with an extremely deterministic and infrequent traffic pattern.

- *Dispersion method*: We defined intermediate Confidence Classes between the one formed by outliers and the one composed by the mass of legit domains. To this end, we use the Upper Half Range[2] (UHR) of the distribution as our dispersion metric and define two new thresholds as $max(CS) - 2 \times UHR$ and $max(CS) - 3 \times UHR$. Based on these thresholds we defined the following Confidence Classes:

- *Low Confidence Class*: formed by domains whose CS falls in the range max(CS) - 3 UHR $>$ CS $\geq$ 25 percentile - 1.5 IQR.
- *Moderate Confidence Class*: formed by domains whose CS falls in the range max(CS) - 2 UHR $>$ CS $\geq$ max(CS) - 3 UHR.
- *High Confidence Class*: formed by domains whose CS falls in the range CS $\geq$ max(CS) - 2 UHR.

Figure 5 shows the four defined Confidence Classes for the December 1, 2016 dataset.

- Predicting the Scoring List: The Scoring list used at day d has to be inferred from a prediction algorithm applied on the historical *Confidence Score* values of domains at days $d-1$, $d-2$, $d-3$, ... We refer to the estimated CS value of a domain X included in this list as $CS^*_d(X)$. To define the prediction algorithm, we first studied the stationary properties of the temporal series of CS values of domains across the 25 days forming our dataset. This analysis revealed that CS values present a high stationarity, with 40 % of the domains in our dataset being strictly stationary (with a 90 % confidence interval), as reported by the Augmented Dickey-Fuller test [43]. The analysis of the autocorrelation and partial autocorrelation functions for these domains revealed that in general, only the CS of the previous day $(CS_{d-1}(X))$ contributes significantly to the prediction of CS(X) at day d. Then, the optimal predictor is $CS^*_d(X) = CS_{d-1}(X)$ and the Scoring List to be used at day d is formed by the $CS^*_d(X)$ of the different domains in our dataset.

As a result of the application of the three described algorithms, the Scoring Module produces each day a Scoring List that includes both the Confidence Score and the Confidence Class for each individual domain.

5.2.4 Filtering Module

This module processes in real-time each received scoring request from the Communication Interface module. In particular, it extracts the domain from the scoring request and searches for the $CS^*_d(X)$ and the Confidence Class associated with the domain in the Scoring list. As a result of this process, the Filtering Module generates a *scoring reply* message including the following information: Bid Request ID (obtained from the corresponding scoring request), the domain's CS and the domain's Confidence Class. The scoring reply is sent to the DSP through the Communication Interface module. The DSP can leverage this information to define its own invalid traffic filtering capability. Note that if the domain extracted from the scoring request is not present in the Scoring list, the scoring reply has the following content <bid request id, NULL, NULL>.

5.3 System Implementation

In this subsection we describe the implementation of Nameles, that meets the performance and scalability requirements defined in Subsection 5.1. For doing this, we used resources with negligible cost in comparison to typical resources available for DSPs and relying in open-source technology.

[2] The UHR is measured as the distance between the median and the maximum value of the CS distribution.

-The Communication Interface and Filtering module: The Communication Interface and the Filtering modules address different aspects of Nameles functionality and thus we have described them separately in Section 5.2. In our Nameles prototype we use an integrated implementation of communication interface and filtering modules for efficiency purposes.

We implement the parallel pipeline communication structure described in Figure 4 on top of ZeroMQ [3] (a highly scalable distributed messaging system implemented in C) using the existing Java bindings for this purpose. On the Nameles side, we use 6 workers that in addition to taking care of the pull and push communication functions, implement the filtering process. Each worker is an independent process, which has an independent copy of the Scoring List hash table produced by the Scoring Module allocated in RAM. Hence, each worker pulls independently scoring requests from the DSP's sending queue. For each scoring request, it extracts the domain id, obtains the CS and Confidence Class associated with the domain from the Scoring List hash table, creates the scoring reply and pushes it to the DSP's receiving queue.

- The Scoring Module:

The Scoring Module implements a temporal hash table including the number of bid requests associated with each pair <domain, IP>. For each new bid request, the counter of the tuple <domain, IP> included in the bid request is increased by 1. To speed up this process, we parallelize it across several workers. At the end of every day, the resulting hash table includes the needed information to compute the Confidence Score for each domain as well as the thresholds to define the different Confidence Classes. For this purpose, we store this temporal table into a PostgreSQL database and use different PostgreSQL functions and Java scripts to obtain the CS and the Confidence Class of each domain. The final result of the process is the Scoring List, which is stored in a PostgreSQL table. For efficiency purposes, we map the Scoring List into a hash table using as a key the domain id and as value the tuple <CS, Confidence Class>. This table is transferred to the ”Communication Interface+Filtering” module to be used in the real-time filtering of bid requests. Finally, the table computed with the data at day d serves as scoring list for day $d + 1$.

6. PERFORMANCE EVALUATION OF THE SYSTEM

We have deployed a realistic experimental set-up to confirm that our Nameles prototype meets the requirements defined in Section 5.1. Specifically, the scalability and delay requirements, and accuracy pertaining to the scoring of domains.

6.1 Experimental Set-up

To conduct the performance evaluation, we have deployed an experimental set-up that replicates a production set-up in actual business use by a large-scale DSP. In particular, we use three servers in our setup for Nameles. The first server plays the role of the DSP. This server reads the bid requests from the CSV files forming our sample dataset and, based on this information, produces a stream of scoring requests to Nameles. The rate of scoring requests is a configurable parameter so that we can perform stress-tests by using significantly higher rates of bids per second than the ones reflected in our dataset. The second server deploys the “Communication Interface and Filtering” module of our Nameles prototype. It receives the stream of scoring requests from the DSP server and processes it to obtain the scoring replies. In addition, this server forwards the scoring requests to a third server, which implements the “Scoring” module.

The server emulating the DSP is a Dell PowerEdge R710 with 16-cores, 48 GB of RAM and 6 TB of hard drive capacity with a non-recurring-cost (NRC) of ∼\$6k. The servers implementing the “Communication and Filtering” and the “Scoring” modules are similar, a Dell PowerEdge R730xd with 24-cores, 64 GB of RAM and 46 TB hard drive capacity with an NRC of ∼\$13 k. Each server is connected to a common 1 Gbps Ethernet switch.

In the context of common use in the Adtech industry, the resources employed in our prototype can be considered commodity hardware. If we assume a depreciation period of 5 years, the monthly cost of such infrastructure is roughly \$500. Based on an average pricing of three common cloud vendors [5, 23, 56], a monthly-recurring-cost of ∼\$1 k is required for a similar configuration in a cloud environment with a zero NRC, further reducing the barrier of entry to adopting Nameles.

6.2 Scalability and Processing Delay

- Scoring List computation time: A critical aspect of the scalability of Nameles resides in its ability to produce the Scoring List in a short time. Specifically, given that the Scoring List is updated daily, the computation process must guarantee that the new list is ready before the expiration of the previous one, i.e., in less than 24 h. We have measured the computation time for the 25 daily datasets, including between 1.7-1.9 B bid requests, and confirmed that the computation time of the Scoring List is always smaller than 4 hours. Hence, Nameles meets the scalability requirements for this critical process.

- Delay and the memory consumption of the filtering process: From the DSP's perspective, the filtering process starts when it sends a Scoring Request and finishes when it receives the corresponding Scoring Reply. In our Nameles prototype this process is implemented by the “Communication Interface+Filtering”

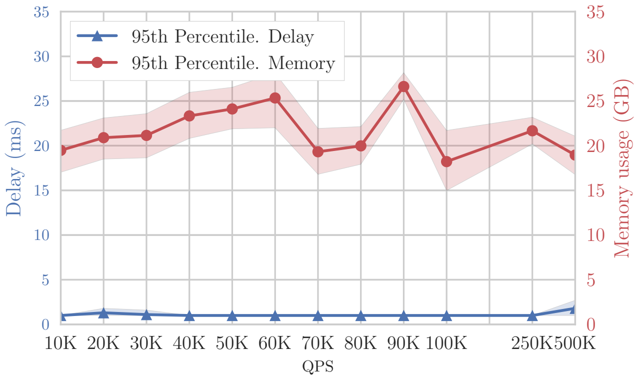

Figure 6: 95 percentile of delay and memory consumption for the filtering process at different input request rates.

module. The analysis of our dataset reveals an average and peak rates of 22 k and 26 k requests per second, respectively. Then, our prototype must meet the following two requirements while processing scoring requests streams at the observed peak rate: not overflowing the memory of the server and offering a small delay to minimize its impact on the aggregate delay of the real-time bidding process.

We have evaluated the performance of our prototype for scoring request streams ranging from 10 k to 500 k queries per second (QPS). For each of the analyzed rates we run stress-tests of 5 minutes. For the case of request rates of 10 k (500 k), these tests generate a total of 3 M (150 M) scoring requests. During the tests, we measure the individual delay associated to the filtering process of each scoring request as well as the overall memory consumption of the filtering process. Figure 6 summarizes the performance of our Nameles prototype. The x-axis shows the different tested scoring request rates. The left y-axis and right y-axis show the 95-percentile filtering delay and 95-percentile memory consumption measured during the experiment for the different scoring request rates (QPS), respectively. Note that each stress test has been run 5 times. The line in the figure represents the average of 95-percentile values across the 5 experiments whereas the lighter color area shows the max and min 95-percentile values.

First of all, we observe that the system performance is quite stable across the different experiments and the observed variability in memory consumption is due to the instantaneous load of the server at the measurement moment rather than the QPS of the experiment. The results of the stress-tests demonstrate that our Nameles prototype offers very high scalability performance. In particular, the 95 percentile of memory consumption and delay are lower than 28 GB and 3 ms for any of the considered QPS. These results prove that our filtering process scales to handle more than 20 B bid requests per day, meeting the requirements of the largest DSPs such as Google, The Trading Desk and MediaMath.

$CS_d(X) \setminus CS^*_d(X)$	No C.	Low C.	Mod. C.	High C.
No Confidence		1.06 %	0.19 %	0.04 %
Low Confidence	0.85 %		1.65 %	0.09 %
Moderate Confidence	0.18 %	1.45 %		2.82 %
High Confidence	0.04 %	0.07 %	2.54 %	

Table 2: Average miss-classification rates among the Confidence Classes for the X days of the dataset.

6.3 Scoring Accuracy

There are two aspects to evaluate with respect to the accuracy of scoring. First, we have to assess the accuracy of our prediction algorithm. Second, we need to assess the accuracy of the Confidence Scores assigned to the domains.

- Accuracy of prediction algorithm: For each of the daily datasets, we have computed the Root Mean Square Error (RMSE) of the difference between the predicted CS ($CS^*_d(X)$) and the actual CS ($CS_d(X)$) across all domains. The results indicate that the RMSE is smaller than 3 points in every case.

In addition, domains are assigned to Confidence Classes, and we have evaluated the miss-classification rate among these classes. Table 2 presents a summary of the average miss-classification rate between each pair of Confidence Classes across the 25 days in our dataset. First of all, we observe that miss-classification rates are below 2.82 % between any pair of classes. A careful analysis of the miss-classified domains indicates that the classification errors are mainly associated with domains having a CS close to the threshold that separates two contiguous classes. This is also coherent with the fact that misclassifications between non-contiguous classes are negligible (< 0.2 %). Finally, the highest miss-classification rates occur between the “Moderate” and “High” Confidence Classes, which are the classification mistakes with the lowest impact for the DSPs business since these are the two classes with higher confidence levels and, in principle, not recommended to be filtered.

- Assessment of Confidence Score accuracy: The accuracy of the Confidence Score cannot be objectively evaluated. There are various continuously changing factors related with the invalid traffic problem; attack vectors, domain traffic profiles, and others. As a result, there are no reliable ground truth datasets available for evaluating invalid traffic filtering solutions. Contrary with propriety verification solutions that suffer from this same issue, Nameles source code can be independently audited. To validate accuracy of the Confidence Scoring, we have worked closely together with the Adtech industry over a period of 18 months to subjectively eval-

		No Conf.		Low Conf.		Moderate Conf.		High Conf.
Alexa Upstream traffic from Google and Facebook (%)	median	20	(−41 %)	18.5	(−45 %)	23.7	(−30 %)	33.7
	IQR	21.05		20.19		24.19		32.84
Alexa Bounce rate (%)	median	41.8	(−27 %)	40.9	(−29 %)	35.3	(−39 %)	57.5
	IQR	32.4		25.6		27.7		28.7
Alexa Search traffic (%)	median	8.1	(−35 %)	7.7	(−38 %)	5.5	(−56 %)	12.5
	IQR	19.7		15.9		16.1		16.9
Alexa Total sites linking to the domain	median	9.2	(−75 %)	131	(−62 %)	256	(−27 %)	348
	IQR	616		371		800		1,198
SimilarWeb Bounce rate (%)	median	51.5	(−12 %)	38.8	(−34 %)	34.9	(−40 %)	58.6
	IQR	24.97		20.84		24.8		24.0
SimilarWeb Direct traffic (%)	median	43.1	(68 %)	34.2	(34 %)	38.1	(49 %)	25.6
	IQR	39.5		37.0		34.6		27.8
SimilarWeb Search traffic (%)	median	21.2	(−31 %)	29.3	(−5 %)	19.5	(−37 %)	30.9
	IQR	39.3		46.5		39.5		39.8

Table 3: Value of external metrics associated with domains in each of the defined Confidence Classes in our dataset.

uate the results provided by Nameles in extensive trials.

Further we performed an assessment using a twofold approach. First, we conducted an analysis that relies on the following metrics, which are extensively used in the Adtech industry to infer the quality of traffic of a domain:

- *Bounce Rate:* This metric measures the fraction of sessions that only visit a single page in a domain. A low bounce rate is a strong indication of low quality traffic.

- *Traffic from popular publishers:* This metric represents the percentage of upstream traffic coming to the domain from popular publishers. In particular, the two publishers contributing a larger fraction of traffic to domains are Google and Facebook. Then, for our validation we will compute the fraction of upstream traffic coming from Google and Facebook to a domain. A very low fraction of traffic coming from Google and Facebook may reveal the presence of low quality traffic.

- *Search Traffic:* This metric measures the percentage of traffic coming to the domain from search engines. A very low search traffic percentage is often an indication of low quality traffic.

- *Direct Traffic:* This metric measures the percentage of traffic that reach the domain directly without being redirected from other website. In this case, a large fraction of direct traffic is usually linked to low quality traffic.

- *Number of sites linking to a domain:* An interesting domain attracting high quality traffic would typically be linked from a large number of other sites. Contrary to this, domains associated to ad fraud or other malicious practices, would typically be linked to from a lower number of sites.

We have queried two well-known services, Alexa [4] and SimilarWeb [45], to obtain these metrics for those domains in our dataset with more than 500 associated bid requests. Note that not all the metrics are offered by both services. Table 3 presents the median and IQR values for the distribution of each one of these metrics for each Confidence Class. In addition, the table shows the relative difference of the median values of these metrics for the "No", "Low" and "Moderate" Confidence Classes in comparison to the "High" Confidence Class.

We observed substantial differences (up to 75 % in some cases) between the "High Confidence" Class and the rest, suggesting that our scoring mechanism is able to accurately identify legitimate domains.

The second approach is based on subjective assessment by industry experts. In particular, we have requested two respected independent research consultants focused on invalid traffic, Dr. Augustine Fou [20] and Mr. Shailin Dhar [39], to assess the accuracy of our scoring system based. They both have provided an endorsement for the system and their public quotes can be found in [40].

Therefore, both the objective analysis based on proxy metrics pertaining the confidence level of a domain as well as the evaluation conducted by individual experts suggest that the accuracy of Nameles' scoring system is suitable for adoption by DSPs.

7. RESULTS OBTAINED FROM NAMELES' EXECUTION

In this section we present the results obtained from applying Nameles to our large-scale dataset. First, we analyze the distribution of domains and traffic across the defined Confidence Classes. Then, we use the corresponding fractions of traffic associated with each Confidence Class as filtering rate input to the economic model described in Section 4 in order to quantify the impact that Nameles may has in the DSP economics based on data from real use.

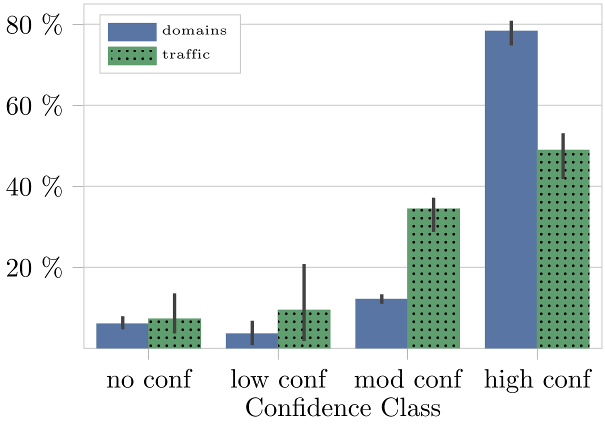

Figure 7: Mean percentages of domains in each of the Confidence Classes across the days of the dataset. The error bars show the minimum and maximum percentage presented.

7.1 Longitudinal Analysis of domains' confidence level

Figure 7 shows the fraction of domains and ad traffic (i.e., bid requests) belonging to each of the defined Confidence Classes for the 25 days in our dataset. The main bar shows the average fraction and the error bar shows the maximum and minimum values across the 25 days in the sample. Note that these results are obtained for domains with at least 500 bid requests in a day in order to guarantee that we have statistically meaningful information about the traffic pattern of the domain. In average (7,3; 9,3; 34,4; 48.9) % of the traffic is associated with ("No", "Low", "Moderate" and "High") Confidence Classes.

In addition, we analyzed how popularity relates to confidence. To this end computed the average (and standard deviation) fraction of traffic within each Confidence Class for domains with at least 500, 1 k, 10 k, 50 k, 100 k and 1 M bid requests per day. Figure 8 shows the results. One may expect that as more popular domains are measured, the fraction of domains within the "High" Confidence Class would increase and the fraction in other groups would decrease. However, we observe the opposite trend, which is emphasized for domains with more than 100 k daily bid requests where we observed how the lines of "Moderate" and "High" Confidence Classes cross.

7.2 Nameles' impact on DSPs' profitability

The results in the previous subsection provides specific figures on the filtering rates that Nameles provides at different confidence level. For instance, a filtering rate of 7.3 % filters out traffic from domains with very rare traffic patterns that offer no confidence. A filtering rate of 16.6 % eliminates traffic offering low or no confidence, and a filtering rate of 51 % filters any domain that does not provide a high confidence.

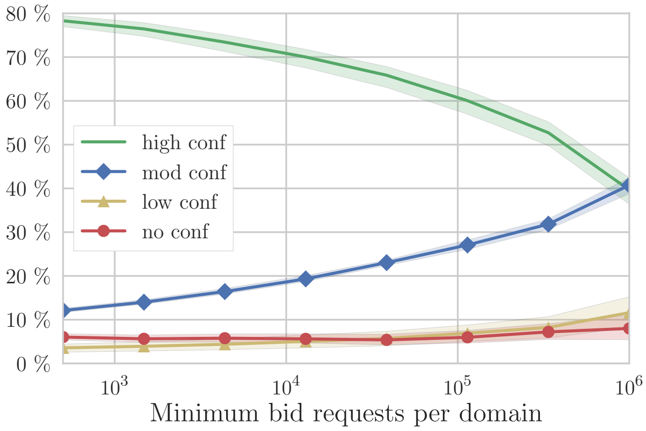

Figure 8: Percentage of traffic in each Confidence Class as function of the minimum bid requests per domain.

Using these filtering rates as input to the economic model presented in Section 4 gives us an estimation of the impact that Nameles is expected to have in the profitability of a DSP. All the limitations associated with the assumptions of the model explained in Section 4 apply here. The obtained results indicate that filtering at the "No Confidence", "Low Confidence" and "Moderate Confidence" level offer NPV (and EV) improvements in comparison to the scenario without filtering of 29, 56 and −105 % (10, 20 and −33 %), respectively. We observe, that filtering at the "Moderate Confidence" level would not be recommended, because Moderate class may include a non negligible fraction of legitimate traffic and will lead to negative economic consequences. On the other hand, filtering at the "No Confidence" or "Low Confidence" class leads to strong positive economic impact.

8. RELATED WORK

In the recent years several studies have unveiled different types of attacks used for generating invalid traffic with the goal of generating monetary gain fraudulently [41, 50, 60], with reported revenues of millions of dollars per day [62]. To address the problem of invalid traffic identification, verification vendors such as Integral Ads Science [28], Double Verify [19], and WhiteOps [61] have emerged in recent years. Also major players of the Adtech industry claim to devote significant attention to address this issue [1]. Unfortunately, all existing commercial solutions are based on opaque proprietary technologies, and it is hard to assess their efficiency in identifying invalid traffic. Some recent studies have proven the inefficiencies of such solutions in identifying even simple invalid traffic attacks [16, 38].

The research community has also addressed the identification of invalid ad traffic. The proposed solutions focus on detecting invalid traffic at the selling side of the online advertising chain, i.e., publishers web pages [15, 49] or delivered ads [11, 25]. These solutions analyze the interaction of the user with the web page or the served ad in order to identify commonly known attacks such as visits generated by bots [11] or redirection attacks [48]. Nameles is the first non-proprietary solution for the identification of invalid traffic to be deployed by the DSP. We conjecture that the lack of such solutions may be caused due to researchers' difficulties in accessing relevant datasets from DSPs and lack of collaboration between DSPs and academic researchers.

From a methodological perspective, there is a previous work that has used entropy to identify invalid video visits to a Chinese video portal [12]. The authors of this paper propose to use entropy as the final metric to assess the traffic quality and a semi-supervised classification that rely on manually labeled samples to differentiate between valid and invalid video traffic. However, as discussed in Section 5, the native Shannon entropy has an important drawback since its interpretation depends on the volume of associated events. To overcome this limitation, we use a Confidence Score based on a normalized version of entropy. Moreover, instead of using manual labeling of suspicious traffic, we define unsupervised statistically supported outlier detection method. Hence, although both papers are based on the same fundamental concept of entropy, the application of this concept is significantly different.

Finally, it is important to highlight that, in contrast to industrial proprietary solutions, Nameles is the first open-source solution for the identification of invalid traffic.

9. NAMELES IN THE REAL WORLD

The opaque model for traffic verification adopted by leading Adtech vendors has a significant drawback, auditing of the solutions is not possible. First research studies in this matter provide evidences that existing solutions present clear deficiencies [16, 38], indicating that opacity may not be the right approach to fight invalid traffic. The current version of Nameles is available on Github with port for PostgreSQL, with resource commitments from two Adtech companies to work on adding ports for Spark and MemSQL, two commonly used database solutions in the Adtech industry.

A key advantage of Nameles is its modularity and simplicity which allow to easy extension, modification and improvement of the platform. For instance, the current implementation of Nameles uses the normalized entropy as the information for identifying invalid traffic, and we acknowledge that this technique is not able to identify all types of invalid traffic. However, the detection module can be extended to include other detection techniques (e.g., Co-Visitation network [49]) to improve the efficiency of the platform. Nameles can be considered a platform for the community-led industry-wide effort to fight invalid traffic in programmatic advertising. As a result of our contribution, the WFA has publicly endorsed Nameles [40]. Moreover, some of the most renowned research consultants in the area of invalid traffic identification have provided endorsements for Nameles and there are already multiple leading Adtech vendor having agreed to trialling Nameles.

10. CONCLUSION

This paper provides for the first time, solid arguments for triggering a paradigm shift in the fight against invalid traffic to a more transparent and community-led direction, and offers a significant addition to the resources available for reducing the burden invalid traffic is creating in the programmatic advertising ecosystem. We do this by showing that intermediaries in the ad supply chain (specifically DSPs) have strong incentives to filter out invalid traffic and by defining and testing a technological solution, Nameles, and show how it can implemented into operational production systems to detect and filter out invalid traffic. We show this delivers DSPs tangible business results such as higher Net Present Value, Enterprise Value, and improved profitability.

Nameles has been released as the first open-source solution for the detection of invalid traffic in programmatic advertising, with the goal of being further improved in a community effort carried on jointly by academic researchers and the industry. The evidenced performance of the current version of Nameles along with our open-source vision has led the World Federation of Advertisers to endorse Nameles as a solution to counter invalid traffic by the Adtech industry. Our effort in the mid-term will be twofold: First, we will collaborate with other contributors to improve Nameles, and second we will work along with WFA and other key advertising industry bodies to support the wide reaching adoption of the Nameles system across the Adtech industry.

References

[1] *AdSense Help. How Google prevents invalid activity.* (Visited on 01/27/2017).

[2] World Federation of Advertisers. *Compendium of ad fraud knowledge for media investors.* 2016.

[3] Faruk Akgul. *ZeroMQ.* Packt Publishing, 2013.

[4] *Alexa: actionable analytics for the web.* URL: http://www.alexa.com (visited on 01/27/2016).

[5] *Amazon EC2 Pricing.* URL: https://aws.amazon.com/ec2/pricing (visited on 01/27/2017).

[6] *An adtech autopsy. A detailed analysis of the Demand Side Platform business model.* URL: http://autopsy.pw (visited on 01/27/2017).

[7] *Botlab*. URL: http://botlab.io (visited on 01/27/2017).

[8] Richard A Brealey et al. *Principles of corporate finance*. Tata McGraw-Hill Education, 2012.

[9] Alexandra Bruell. *Inside the Hidden Costs of Programmatic*. URL: http://adage.com/article/print-edition/inside-hidden-costs-programmatic/300340 (visited on 01/27/2017).

[10] Internet Advertising Bureau. *Transparency is the key to programmatic success*.

[11] Patricia Callejo et al. "Independent Auditing of Online Display Advertising Campaigns". In: *Proceedings of ACM Hotnets*. 2016.

[12] Liang Chen, Yipeng Zhou, and Dah Ming Chiu. "Fake view analytics in online video services". In: *Proceedings of Network and Operating System Support on Digital Audio and Video Workshop*. ACM. 2014.

[13] Media Rating Council. URL: http://mediaratingcouncil.org (visited on 01/27/2017).

[14] Media Rating Council. *Invalid Traffic Detection and Filtration Guidelines Addendum*. URL: http://mediaratingcouncil.org/101515_IVT%20Addendum%20FINAL%20(Version%201.0).pdf (visited on 01/27/2017).

[15] Vacha Dave, Saikat Guha, and Yin Zhang. "Viceroi: catching click-spam in search ad networks". In: *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. ACM. 2013.

[16] Shailin Dhar. *Mystery Shopping Inside the Ad Fraud Verification Bubble*. http://www.slideshare.net/ShailinDhar/mystery-shopping-inside-the-adverification-bubble. 8, 2016.

[17] Digiday. *WTF is programmatic?* URL: http://digiday.com/wp-content/uploads/2016/07/WTF_programmatic-2016.pdf (visited on 01/27/2017).

[18] Distil. *Distil Networks Releases New Data on The State of Digital Advertising Fraud*. URL: https://resources.distilnetworks.com/press-releases/distil-networks-releases-new-data-on-the-state-of-digital-advertising-fraud (visited on 01/27/2017).

[19] *Double Verify*. URL: http://www.doubleverify.com (visited on 01/27/2017).

[20] *Dr. Augustine Fou*. URL: https://www.linkedin.com/in/augustinefou (visited on 01/27/2017).

[21] Emarketer. *The Ad Industry's Focus on Fraud Has Intensified*. 9, 2016. URL: https://www.emarketer.com/Article/Ad-Industrys-Focus-on-Fraud-Has-Intensified/1014430 (visited on 01/27/2017).

[22] Google. *DoubleClick RTB Protocol. Latency Restrictions and Peering*. 11, 2017. URL: https://developers.google.com/ad-exchange/rtb/peer-guide.

[23] *Google Cloud Platform Pricing Calculator*. URL: https://cloud.google.com/products/calculator (visited on 01/27/2017).

[24] *Gurufocus*. URL: http://www.gurufocus.com (visited on 01/27/2017).

[25] Hamed Haddadi. "Fighting online click-fraud using bluff ads". In: *ACM SIGCOMM Computer Communication Review* (2010).

[26] Michi Henning and Mark Spruiell. "Distributed programming with ice". In: *ZeroC Inc. Revision* (2003).

[27] Sarah Herrod. *D́igital Spend Will Be More Than 25% In 2016Ṕredicts Dentsu Aegis*. URL: http://www.bandt.com.au/media/digital-spend-will-25-2016-predicts-dentsu-aegis (visited on 01/27/2017).

[28] *Integral Ad Science (IAS)*. URL: https://integralads.com (visited on 01/27/2017).

[29] Investopedia. *DCF Analysis: Coming Up With A Fair Value*. URL: http://www.investopedia.com/university/dcf/dcf4.asp (visited on 01/27/2017).

[30] Investopedia. *Rate of Return*. URL: www.investopedia.com/terms/r/rateofreturn.asp.

[31] Isba. *Ad Fraud and Ad Blocking*. URL: http://www.isba.org.uk/news/2016/01/14/ad-fraud-and-ad-blocking (visited on 01/27/2017).

[32] *Joint Industry Committee for Web Standards (JICWEBS)*. URL: https://jicwebs.org (visited on 01/27/2017).

[33] Schubert Jonckheer & Kolbe. *Rocket Fuel Executives Under Investigation*. 16, 2016. URL: http://www.classactionlawyers.com/blog/2016/2/16/rocket-fuel-executives-under-investigation (visited on 01/27/2017).

[34] IAB Technology Laboratory. *Real Time Bidding (RTB) Project. OpenRTB API Specification Version 2.5*.

[35] Stephan Loerke. *Lack of scrutiny has given ad fraud criminals a head start.* 2, 2016. URL: http://www.campaignasia.com/article/lack-of-scrutiny-has-given-ad-fraud-criminals-a-head-start/405627 (visited on 01/27/2017).

[36] Investopedia. J.B. Maverick. *What is the long-term average growth rate of the telecommunications sector?* URL: http://www.investopedia.com/ask/answers/071515/what-longterm-average-growth-rate-telecommunications-sector.asp (visited on 01/27/2017).

[37] Digiday. John McDermott. *Terry Kawaja: 'Winter is coming' for the ad tech industry.* URL: http://digiday.com/platforms/terry-kawaja-winter-coming-ad-tech-industry (visited on 01/27/2017).

[38] Marciel Miriam et al. "Understanding the Detection of View Fraud in Video Content Portals". In: (2016).

[39] *Mr. ShailinDhar.* URL: https://www.linkedin.com/in/shailindhar (visited on 01/27/2017).

[40] *Nameles - Open source invalid traffic detection.* URL: http://nameles.org (visited on 01/27/2017).

[41] Paul Pearce et al. "Characterizing large-scale click fraud in zeroaccess". In: *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2014.

[42] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. New York, NY, USA: John Wiley & Sons, Inc., 1987.

[43] Said E Said and David A Dickey. "Testing for unit roots in autoregressive-moving average models of unknown order". In: *Biometrika* (1984).

[44] C. E. Shannon. "A mathematical theory of communication". In: *The Bell System Technical Journal* (1948).

[45] *SimilarWeb: Website Traffic & Mobile App Analytics.* URL: https://www.similarweb.com (visited on 01/27/2016).

[46] Smaato. *Mobile RTB Insights Report Q3 2014.* URL: https://www.smaato.com/resources/reports/mobile-rtb-insights-q3-2014 (visited on 01/27/2017).

[47] *Snort - Network Intrusion Detection & Prevention System.* URL: https://www.snort.org (visited on 01/27/2016).

[48] Kevin Springborn and Paul Barford. "Impression Fraud in On-line Advertising via Pay-Per-View Networks". In: *22nd USENIX Security Symposium (USENIX Security 13)*. USENIX, 2013.

[49] Ori Stitelman et al. "Using co-visitation networks for detecting large scale online display advertising exchange fraud". In: *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2013.

[50] Brett Stone-Gross et al. "Understanding Fraudulent Activities in Online Ad Exchanges". In: *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*. ACM, 2011.

[51] *Storm, Apache.* URL: http://storm.apache.org (visited on 01/27/2017).

[52] The Telegraph. *Matomy shares battered in digital ad fraud crackdown.* 23, 2015. URL: http://www.telegraph.co.uk/finance/newsbysector/mediatechnologyandtelecoms/digital-media/11558623/Matomy-shares-battered-in-digital-ad-fraud-crackdown.html (visited on 01/27/2017).

[53] *The Bro Network Security Monitor.* URL: https://www.bro.org (visited on 01/27/2017).

[54] Financial Times. *Google charges for YouTube ads even when viewed by robots.* URL: https://www.ft.com/content/f9da727c-6207-11e5-9846-de406ccb37f2 (visited on 01/27/2017).

[55] *Trustworhy Accountability Group (TAG).* URL: https://tagtoday.net (visited on 01/27/2017).

[56] UpCloud. *Pricing.* URL: https://www.upcloud.com/pricing (visited on 01/27/2017).

[57] Vizeum. *Mobile to drive digital spend increase to 25% of total ad spend in 2016.* URL: http://vizeum.co.uk/p/news-item/mobile-to-drive-digital-spend-increase-to-25-of-total-ad-spend-in-2016 (visited on 01/27/2017).

[58] Joint Industry Committee for Web Standards. *Traffic Fraud: Best Practices for Reducing Risk to Exposure*. 2015.

[59] WFA. *WFA guide to Programmatic Media.* URL: http://www.wfanet.org/media/programmatic.pdf (visited on 01/27/2017).

[60] WhiteOps. *The Methbot Operation*. 20, 2016.

[61] *Whiteops.* URL: https://www.whiteops.com (visited on 01/27/2017).

[62] ANA & WhiteOps. *2015 Bot Baseline: Fraud in Digital Advertising*. 2016.

[63] *World Federation of Advertisers.* URL: http://www.wfanet.org/en/about-wfa/what-is-the-wfa (visited on 01/27/2017).

[64] IMPERVA INCAPSULA. Igal Zeifman. *Bot Traffic Report 2016.* URL: https://www.incapsula.com/blog/bot-traffic-report-2016.html (visited on 01/27/2017).

[65] Weinan Zhang, Shuai Yuan, and Jun Wang. "Optimal real-time bidding for display advertising". In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2014.